IMPROVED MOBILE ROBOT LOCALIZATION USING SEMANTIC WORLD MODELS

Bjoern Sondermann, Markus Emde, and Juergen Rossmann

Institute for Man-Machine Interaction, RWTH Aachen University, 52074 Aachen, Germany

ABSTRACT

In mobile robotics, simultaneous localization and mapping is a complex problem. However, by using smart constraints, the problem can be reduced considerably. Instead of constraining the issue to a specific robotic system or its movement behavior, we show how semantic environment perception and modeling allows another point of view and therefore a convenient solution for the problem. We present a method for application-independent localization and mapping based on semantic landmarks and the concept of visual odometry. Central starting point is a generic landmark definition, allowing for a reduction of the 3d localization problem to a more efficient search for an affine transformation in 2d space. These semantic landmarks are used to simultaneously map the surrounding environment of the robot, resulting in a widely applicable world model.

Key words: Mobile Robotics; Localization; Semantic World Modelling.

1. INTRODUCTION

In our previous work, we focused on the localization of mobile robots in exploration missions using navigation maps that already existed or were generated during the landing phase of the mission [10]. However, you cannot always rely on a perfect landing at the planned landing site (see project Rosetta/Philae). Thus, if no information of the actual landing region exists, it is necessary that the rover perceives its environment, determines its movement and simultaneously generates a navigation map.

Based on the ideas of well-known Visual Odometry approaches, connected with the concept of semantic landmarks, we extended our existing localization framework by a new localization and mapping approach. The navigation maps generated by our SLAM-algorithm allow to use our existing, robust and verified algorithms for localization without further effort. At the same time, the results of the SLAM algorithm can also be used to refine the navigation map. It does not matter whether the original map was created in advance or during the exploration. To verify the results, two prototypes of self-contained localization units have been designed and built which include processing units and the necessary sensors (stereo camera, laser scanner, IMU). In first test series, both prototypes have already been used for verification of the localization concept, i.e. as shown in [4] in a terrestrial application scenario. The introduced SLAM approach has been successfully verified using recorded data of these tests. Furthermore, a transfer to planetary exploration missions has already been carried out in our Virtual Space Robotics Testbed [13] and is presented in this paper.

By adding a mapping component to our localization approach [4] the framework now is application-independent and can be used in further application scenarios. These are presented in the outlook.

In the following section, existing localization and mapping approaches are presented which resulted in our approach of an application-independent localization strategy based on semantic landmarks. Section 3 introduces the localization framework in detail as well as environment perception and the semantic world model. Furthermore, the implemented localization and mapping approaches are described in detail. Section 4 lists results from the carried out tests.

2. RELATED WORK

Determining the position and orientation of a mobile platform is an important problem of robotics in different scenarios and has been handled in many different ways. In the majority of cases, optical sensors are used to gather information of the environment and to detect features in the sensor data. These features are sensor dependent attributes, like distance jumps in laser scanner data or corners in image data. By detecting features in subsequent data recordings or *frames* and matching them, the proper motion of the sensor can be reconstructed [8, 9, 7, 2]. The main disadvantage of these approaches is the weak reliability of the features over many frames and the moderate uniqueness for definite matchings. Thus, techniques like visual odometry are prone to errors, which accumulate during runtime, resulting in large drifts. To manage these errors in advance, many constraints are introduced for restricting the problem either to one specific mobile robot

platform and its movement behavior or to the sensor device in use. Furthermore, few approaches restrict both (like [15, 1]).
In our approach, we do not use features in the sensor data for estimating correspondences in subsequent frames. Instead, the sensor data is used to perceive the environment and to semantically describe it. Thus, we get a semantic model of the surrounding environment, which is used to build a world model according to an initially given starting point. This world model simultaniously is used to localize the mobile robot. The available sensor data is analyzed and objects are detected that will be used as landmarks for localization and mapping. There are quite a few publications using the term *landmark* for localization [2, 5, 6, 16], but they do not describe any recognizable objects. Instead, those landmarks are specific features in sensor data that have been processed in any way.
In our approach, the specific objects used as landmarks depend on the application scenario and can be natural objects like trees and rocks in outdoor environments, or artificial markers for indoor applications. To be independent from a particular application, we use a generic landmark definition throughout the presented approach, so that it can be used for any scenario without further adaptations. Only the set of objects used as landmarks and the sensor(s) used to detect them have to be defined (see section 3).
Starting point of this approach was the development of a global localization method for vehicles in forestry. Here, trees served as natural landmarks and laser scanners and stereo cameras were used as primary sensors. The *VisualGPS* called approach was highly accurate and also capable of managing the kidnapped robot problem [14, 4]. However, the global localization method needed a navigation map in advance to operate in its area. This map was previously generated by remote sensing data or by manual recordings. A major disadvantage was that it was not possible to localize the vehicle in unknown environments without a given map.
By integrating a SLAM-algorithm, navigation maps are now generated simultaneously and allow to use our existing, robust and verified algorithms for localization without further effort or adoptions. This is highly relevant i.e. for space missions where no navigation maps of the surrounding terrain of the mobile robot are present or cannot be processed by remote sensing.

3. THE LOCALIZATION FRAMEWORK

The idea of a self-localization and navigation unit was firstly introduced in [11]. It consists of three individual parts: The first part is the sensor control and sensor data pre-proceeing module handling the communication to the sensor hardware and allowing for simple sensor data filtering. It is presented in detail in [3]. The environment perception and semantic world modelling module is the central element of our application-independent localization framework, as the localization itself is carried out on the resulting semantic objects or so-called semantic landmarks. This localization module constitutes the third part of the localization framework. Figure 1 sums up the inner structure of this framework, which will be presented in detail below.

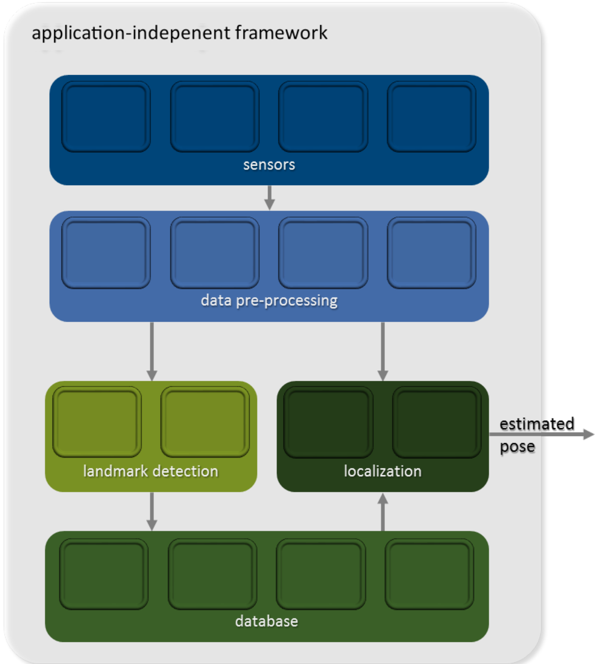

Figure 1. Flow-chart of the presented application independent localization concept

3.1. Environment Perception and Semantic World Modeling

Our localization concept is based on local environment perception as well as on semantic navigation maps as presented in [10]. The localization uses an abstract landmark definition as central data type, with two specializations: a *landmark* and a *perception*. Figure 2 shows a short excerpt of the used data structure. A navigation map consists of semantic objects, that contain a position and a reliability value, indicating the certainty of the classification algorithm when the object was observed. These objects are used as landmarks, allowing us to orientate ourself on the basis of their positions. Perceptions are semantic objects that are currently detected in the sensor data. They contain a direction and a distance which is equivalent to a position in the local reference system of the sensor at the moment of acquisition. The specific inheritances of landmarks and perceptions contain additional attributes describing the particular semantic characteristics of the objects.

For localization of mobile systems, objects are required, that are quantifiable and spatial restricted for the eligibility as landmarks. Which objects should be detected depends on the application and has to be defined at first. For localization, the landmarks have to be reliably detectable and their distance to the sensor has to be de-

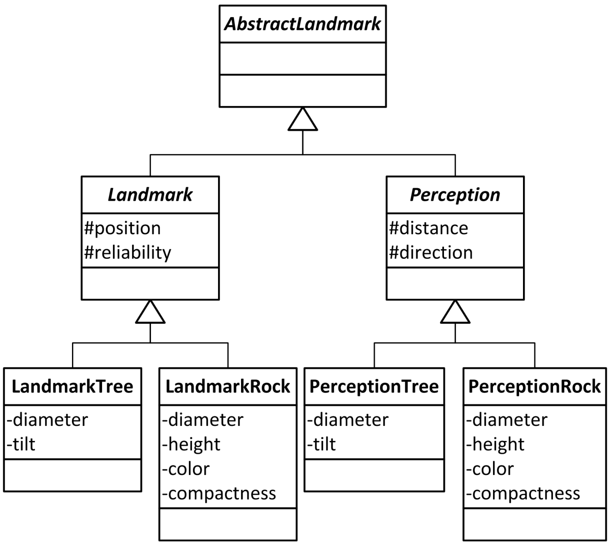

Figure 2. Example for the landmark class hierarchy in UML

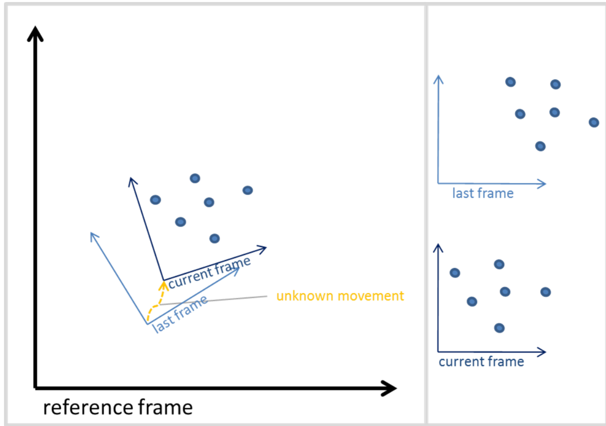

Figure 3. Relations among the used frames; left: current frame and last frame as well as landmarks from map according to the given reference frame; right: landmarks from map according to known last frame and currently perceived landmarks in unknown current frame.

termined. Thus, LIDAR-sensors, as well as stereo and time-of-flight cameras are perfectly suited for landmark detection. When using artificial landmarks - like optical markers - a simple monocular camera can also be used, assuming the dimensions of the markers are known and therefore the distance can be determined from the projected image dimensions and the intrinsic camera parameters.

For the detection of individual landmark types, an appropriate detector has to be implemented for each sensor class, for example for laser scanners and stereo cameras, as the data of these sensors are very different and not quite comparable. After the detection of objects and describing them semantically, it is unimportant which sensor was used to detect them, as the localization is carried out in the semantic world model. To put it simple, the environment perception and modeling module serves as a big sensor fusion funnel, gathering the data of many different sensors and fusing them to one consistent world model of semantic objects. As an example we presented in [4] a rock detection algorithm working on depth data as given by time-of-flight or stereo camera systems.

3.2. Particle-filter-based Localization

After detecting semantic objects in the surrounding environment they can be used as landmarks for localization. If a map of landmarks is preliminary given, for example generated by the techniques proposed in [10], a simple particle filter-based algorithm can be used for localization. In our previous work, we used a method based on [17], adjusting the positions of the locally observed perceptions with the positions of the landmarks in the navigation map. An adaptation of this algorithm to a scenario-independent localization concept was firstly proposed in [11]. The algorithm evaluates randomly distributed pose hypotheses by comparing the position, type and characteristics between the perceptions and the landmarks of the navigation map. If an initial position of the exploring robot is known via satellite data or by knowing the landing spot of a planetary exploration mission, the radius of the particles that are distributed from the algorithm can be reduced for quicker and more reliable initialization.

The downside of this robust and accurate approach is the need for a detailed navigation map in advance which cannot be assured on every mission. In this case an alternative localization strategy has to be used, which is independent from preliminary map data but capable of a reliable pose estimation on semantic objects. A simple approach based on feature tracking in the sensor data would not use this kind of information but is quite error prone and does not allow for example the recognition of already visited places, leading to large deviations over time. A better way is to use the semantic environment perception from above and add a mapping component for building the navigation map on our own.

3.3. Landmark-based Localization and Mapping

The leading advantage of using landmarks in a semantic world model instead of features in sensor data is the reduction of the localization problem into a much simpler problem. As the perceived landmarks from sensor data need to be compared to the landmarks in the map, the solution of the localization problem is a simple affine transformation. That is, the transformation of the landmarks perceived from the last known robots pose into the landmark map is estimated as this represents the movement of the robot.

Obviously, there is no scaling involved as the distances among the landmarks stay fixed. A shear mapping is not necessary, so that a transformation consists of a rotation

and a translation.
Geometrically, the problem can be stated as following: The initially given pose of the robot defines the frame of reference, which is also used by the landmark map. The initial pose of the robot can be determined by calibrating the starting position of the robot with GNSS receivers or checkpoints. If the map is empty and there is no need for a geo-referenced map the reference frame can be determined freely. In this case, the origin will be chosen.
The last known pose of the robot will be denoted by *last frame*. As this is a relative localization method, the starting position has to be given according to the reference frame. In the next steps, the movement from one time step to the subsequent one will be determined and stored as *last movement*. Afterwards, the *current frame* will be updated by applying the movement to the last frame. Figure 3 illustrates the relations among the single frames. The current frame will be determined after landmark detection within three steps:

1. Search for corresponding landmarks in the map
2. Estimate the robots movement and update the *current frame*
3. Update the landmark map

Finding corresponding landmarks In the first step, a possible match for every locally perceived landmark will be searched in the landmark map. This is the most important part of localization as a good motion estimation requires correct matches between observations and the reference map. The search can be carried out in two different reference systems: in the reference frame or in the current frame. The current frame is not known at this time, because this is what we are trying to estimate right now. Thus, the landmark positions from the map cannot be transformed into the reference frame as well as the perceived landmarks cannot be transformed from the current frame into the reference frame. But as the last movement is stored, we can make an assumption by applying the last movement to the last frame again, and call this *proposed frame*. Based on the proposed frame matching landmarks from the map can be searched within a given radius. For calculating with rather small numerical values, the landmarks from the map should be transformed into the proposed frame. This leads to more stable solutions in the movement estimation step.
As rotations cause increasing displacements between local and global landmarks with higher distances to the rotation center, the search space has to be adjusted accordingly. This is important when the robot starts rotating faster. A simple but well working solution is to execute the matching more often with different proposed frames obtained by changing the rotation value from last movement stepwise at certain amounts. The matching with the lowest mean deviation of displacement among the matched landmarks will be taken.

Estimating the robots movement The movement from the last frame to the current frame of the robot has to be stated as movement from the landmarks from the landmark map L given in the last frame into the matching landmarks perceived from the sensor data P given in current frame coordinates. This transformation is given by

$$Rl + t = p,$$

for all landmark positions $l \in L$ and $p \in P$, the rotation matrix

$$R = \begin{pmatrix} cos(\alpha) & -sin(\alpha) \\ sin(\alpha) & cos(\alpha) \end{pmatrix}$$

and the translational vector $t = (u, v)^T$. This transformation can be determined using a least squares approach. Thereto, the equation has to be restated as a linear equation system of the form $Ax = b$, with

$$A = \begin{pmatrix} p_x^1 & -p_y^1 & 1 & 0 \\ p_y^1 & p_x^1 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ p_x^n & -p_y^n & 1 & 0 \\ p_y^n & p_x^n & 0 & 1 \end{pmatrix},$$

with p_x^i and p_y^i the x- or y-value of the i-th perceived landmark, with $i \in 1..n$ and n the number of matches between perceived landmarks $p \in P$ and landmarks from the map $l \in L$. When b is given as

$$b = (l_x^1, l_y^1, \cdots, l_x^n, l_y^n)^T,$$

the solution of the equation system is $x \in \mathbb{R}^4$ with

$$x = (cos(\alpha), sin(\alpha), u, v)^T.$$

Updating the landmark map Given the estimation of the current frame, the landmark map can be updated. Therefore, the perceived landmark positions are transformed into the current frame and with it into the reference system of the landmark map. Perceived landmarks without an existing match within a given radius inside the map will be inserted as new. By adding a *sighting counter* to each landmark of the map indicating how often the landmark has been seen, the landmarks position can be updated by the mean value of all its sightings. This can be done for the landmarks attributes if available.

The resulting map can be used directly without further adaptations by the particle filter-based localization approach described above, allowing also the parallel use of both approaches simultaneously. When some landmarks were added to the map, the particle filter-based localization algorithm provides more reliable position estimation on fast and large movements between two frames. When the movement is quite slow, the affine transformation is calculated much faster, so a combination of both approaches results in a fast, accurate and highly reliable simultaneous localization and map building.

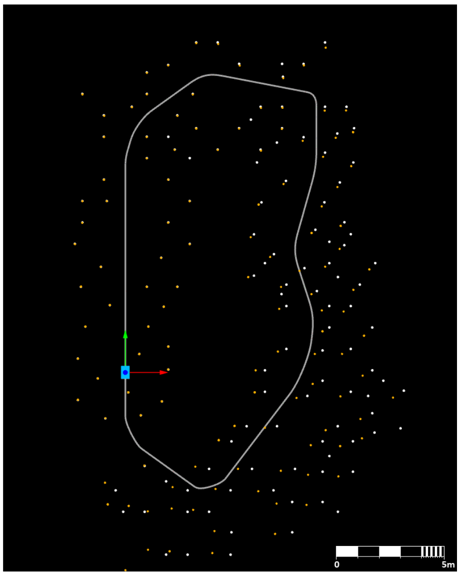

Figure 4. Simulated test run: Landmarks (white dots) are randomly distributed arround the track. The generated landmark map is represented in brown dots.

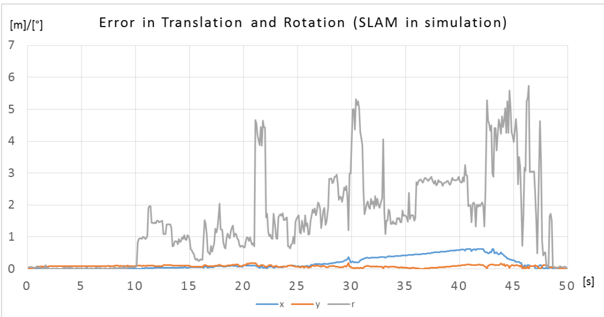

Figure 5. Errors during test run measured against ground truth from simulation. The displacement errors x and y are given in meters and the rotational error r is given in degrees.

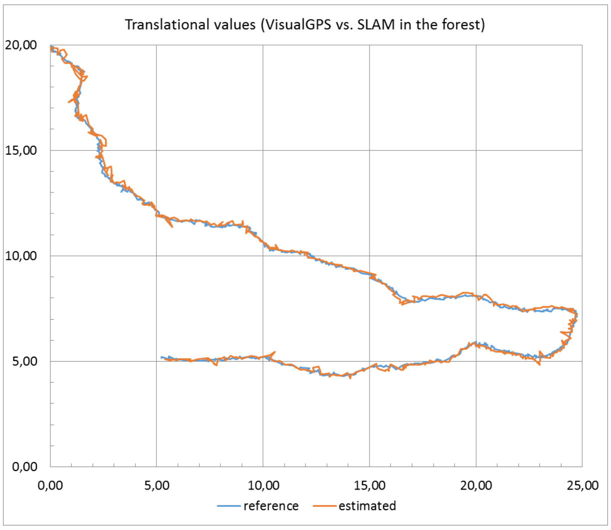

Figure 6. Comparison between results from global localization method and the proposed localization approach.

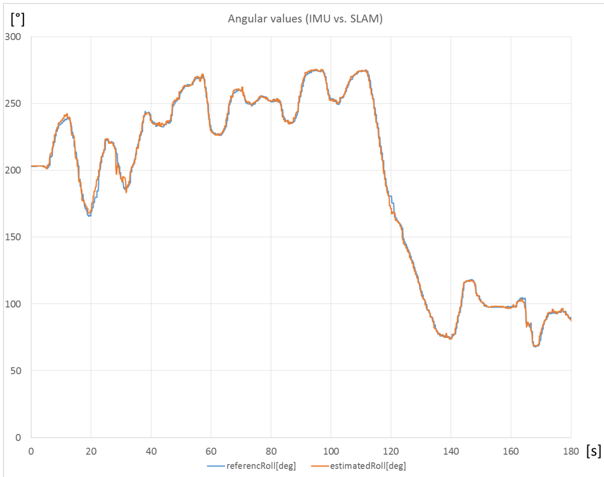

Figure 7. Comparison between rotation values measured by an IMU and the estimated values from the proposed localization approach.

4. RESULTS

The accuracy of the presented simultaneous localization and mapping (SLAM) approach has been tested regarding two criteria. On the one hand the quality of the localization has been determined and on the other hand the accuracy of the resulting map has been evaluated. The analyses have been carried out in a 3d simulation system, as well as on a test site using physical sensor hardware on a robot. For landmark detection we used the 2d laser scanner LMS151 from SICK in the forst respectively a simulated counterpart in simulation. For evaluation purposes in the forest the inertial measurement unit (IMU) IG500A from SBG-Systems has been applied.

In the simulation, we have direct access to ground truth data, so that the estimated movement, as well as the established landmark map can be directly assessed. Figure 4 shows the analysis of the test scenario in the simulation. It was a round course with randomly generated objects, which were used as landmarks for localization and mapping. The image shows the ground truth data of the scenario including the exact movement track of the robot (small blue box with coordinate axes), and the (white) landmark positions. The generated navigation map is visualized as brown dots. Figure 5 shows the error over time for the axis x and y in meters as well as the rotational error r in degrees. A certain drift of up to $0.5m$ can be recognized over time, but as the robot approaches the start position again (at $t = 45s$), the movement is automatically corrected, when the landmarks from the beginning are observed again.

Figure 8. Self-contained localization unit mounted on mobile robot.

The analysis of the test run on a site with physical sensors attached to the robot shown in figure 8 was more challenging as there was no ground truth available directly. Therefore, the landmark positions had to be measured in advance. Afterwards, two tests were executed. At first, the localization component of the proposed approach was tested separately by using the landmark map given. That means, no new landmarks were added during this test run. The estimated rotation has been compared to the data recorded from the IMU. The start position was measured manually and the movement was compared to the results of the demonstrably dependable global localization method VisualGPS. The results are given in Figures 6 and 7. In the second test run, the landmark map was constructed by the SLAM-approach itself.

4.1. Self-Contained Localization Units

Based on the application-independent localization approach previously introduced, two different prototypes of self-contained localization units were designed and manufactured. They are intended for use in outdoor and harsh environments and can, among other carrier systems, be mounted on mobile robots, cars or work machines or carried in hand. The first prototype (as shown in figure 8) weighs 8.7 kg without power supply in its current configuration while having a size of 358 x 314 x 121mm (HWD). Already integrated in this localization unit are an industrial laser scanner and a stereo camera as the primary sensors. In addition, an inertial measurement unit (IMU) with three orthogonally disposed acceleration sensors and three gyroscopes is included. The sensor data processing, mapping and localization are performed on an industrial PC, which is mounted in the same enclosure.

The second prototype uses the same type of sensors, but is trimmed to a minimum of weight. Including all components (sensors, power supply, tablet PC) it weighs only 2.5 kg while having a size of 110 x 300 x 340mm (HWD) including a rack to mount the tablet PC. Figure 9 shows the second prototype without the rack to carry the tablet PC.

Figure 9. Second prototype of a self-contained localization unit designed to be carried by a person or robot with limited load.

Both location unit have external interfaces to connect additional sensors which can be used directly by the localization framework. Furthermore, the self-contained localization units can be operated via a protocol provided by the simulation system VEROSIM – ”Virtual Environments and RObot SIMulation” [12]. The sensor data as well as the results of the data processing algorithms can be recorded at any point in the processing chain in order to perform a later analysis. For data transmission, a wired or wireless communication can be used. Furthermore, the landmark detectors can be reconfigured and replaced during operation as well. This reconfigurability allows the use of the self-contained localization unit in new areas of application.

5. OUTLOOK

The presented approach for landmark-based localization and mapping offers a sound foundation for enhancements and applications in further areas. By using a generic landmark concept for semantic environment perception and modeling many additional use cases are possible.
The developments for landmark based localization and mapping began in the forest and will further be used as foundation for future developments in this area. One concrete use case is the automatic acquisition of forest stands by mapping the tree positions and diameters and the derivation of relevant attributes for the forestry. A further use case is the generation of new forest track networks whose exclusive use is prescribed for heavy machinery. The presented localization and mapping concept can support the driver by indicating deviations of the originally planed tracks.
Outside the forest, there are also many possible applications for the presented approach. Particularly, for indoor localization of automated guided vehicles (AGV) or cars in underground parking or multi-storey car park the determination of an exact pose and movement is required. In these cases the use of optical markers as artificial landmarks is feasible without much additional

effort. A combination of 2d laser scanners and reflection markers provide solid landmarks with a precise detection rate. The map can be generated by the vehicle itself and new markers can be used later to indicate hazardous areas or movable goods.

ACKNOWLEDGMENTS

Parts of this work were developed in the context of the research projects Virtual Forest, SELOK, FastMap and ViTOS.
Virtual Forest is co-financed by the European Union and the federal state of North Rhine-Westphalia, European Regional Development Fund (ERDF). Europe - Investing in our future.
SELOK/FastMap/ViTOS were supported by German Aerospace Center (DLR) with funds of the German Federal Ministry of Economics and Technology (BMWi), support code 50 RA 0911 (SELOK), 50 RA 1034 (FastMap) and 50 RA 1304 (ViTOS).

REFERENCES

[1] S. Choi, J. Joung, W. Yu, and J. Cho. What does ground tell us? monocular visual odometry under planar motion constraint. In *International Conference on Control, Automation and Systems (ICCAS)*, pages 1480–1485, 2011.

[2] M. N. Dailey and M. Parnichkun. Landmark-based simultaneous localization and mapping with stereo vision. In *Proc. Asian Conference on Industrial Automation and Robotics, Bangkok, Thailand*. Citeseer, 2005.

[3] M. Emde, J. Rossmann, B. Sondermann, and N. Hempe. Advanced Sensor Simulation In Virtual Testbeds : A Cost-Efficient Way To Develop And Verify Space Applications. In *AIAA Space 2011 American Institute of Aeronautics and Astronautics (AIAA) Confrence and Exposition*, pages 1–11, 2011.

[4] M. Emde, B. Sondermann, and J. Rossmann. A Self-Contained Localization Unit for Terrestrial Applications and Its Use in Space Environments. In *The 12th International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS)*, 2014.

[5] S. Frintrop and P. Jensfelt. Attentional Landmarks and Active Gaze Control for Visual SLAM. *IEEE Transactions on Robotics*, 24(5):1054–1065, Oct. 2008.

[6] S. C. Hirtle. Landmarks for navigation in human and robots. In *Robotics and Cognitive Approaches to Spatial Mapping*, pages 203–214. Springer, 2008.

[7] K. Konolige, M. Agrawal, and J. Sola. Large-scale visual odometry for rough terrain. *Robotics Research*, 2011.

[8] O. Naroditsky, X. S. Zhou, J. Gallier, S. I. Roumeliotis, and K. Daniilidis. Two efficient solutions for visual odometry using directional correspondence. *IEEE transactions on pattern analysis and machine intelligence*, 34(4):818–24, Apr. 2012.

[9] D. Nistér, O. Naroditsky, and J. Bergen. Visual odometry for ground vehicle applications. *Journal of Field Robotics*, 23(1):3–20, Jan. 2006.

[10] J. Rossmann, G. Jochmann, and F. Bluemel. Semantic Navigation Maps for Mobile Robot Localization on Planetary Surfaces. In *12th Symposium on Advanced Space Technologies in Robotics and Automation (ASTRA 2013), Session 9 Navigation & Localisation II, 15 - 17 May*, pages 1–8. ESA / ESTEC, ESA, 2013.

[11] J. Rossmann, C. Schlette, M. Emde, and B. Sondermann. Discussion of a Self-Localization and Navigation Unit for Mobile Robots in Extraterrestrial Environments. In *The 10th International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS)*, pages 46–53, 2010.

[12] J. Rossmann, M. Schluse, C. Schlette, and R. Waspe. Control by 3D Simulation - A New eRobotics Approach to Control Design in Automation. In C.-Y. Su, S. Rakheja, and H. Liu, editors, *Proceedings of the 5th International Conference on Intelligent Robotics and Applications (ICIRA 2012)", October 3-5, 2012, Montreal, Quebec, Canada*, volume Part II of *LNAI 7507*, pages 186–197, 2012.

[13] J. Rossmann, M. Schluse, B. Sondermann, M. Emde, and M. Rast. Advanced Mobile Robot Engineering with Virtual Testbeds. In *Proceedings for the Conference of ROBOTIK 2012, 7th German Conference on Robotics, May 21-22, Munich*, pages 331–336. VDE Verlag GmbH Berlin, 2012.

[14] J. Rossmann, B. Sondermann, and M. Emde. Virtual Testbeds for Planetary Exploration: The Self-Localization Aspect. In *11th Symposium on Advanced Space Technologies in Robotics and Automation (ASTRA 2011)*, 2011.

[15] D. Scaramuzza, F. Fraundorfer, and R. Siegwart. Real-time monocular visual odometry for on-road vehicles with 1-point RANSAC. *2009 IEEE International Conference on Robotics and Automation*, pages 4293–4299, May 2009.

[16] S. Thrun. Finding landmarks for mobile robot navigation. In *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*, volume 2, pages 958–963. IEEE, 1998.

[17] S. Thrun, W. Burgard, and D. Fox. Probabilistic robotics. 2005, 2005.